



# Red Hat Fuse 7.6

## 마이그레이션 가이드

Red Hat Fuse 7.6으로 마이그레이션



# Red Hat Fuse 7.6 마이그레이션 가이드

---

Red Hat Fuse 7.6으로 마이그레이션

## 법적 공지

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 초록

이 가이드를 사용하여 Fuse 설치를 최신 Red Hat Fuse 버전으로 업그레이드할 때 도움이 됩니다.

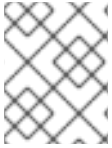
## 차례

머리말 .....	3
<b>1장. FUSE ONLINE 업그레이드 .....</b>	<b>4</b>
1.1. OPENSIFT ONLINE에서 실행 중인 FUSE ONLINE 통합 업그레이드	4
1.2. OCP에서 FUSE ONLINE 업그레이드	4
<b>2장. SPRING BOOT 독립형에서 FUSE 애플리케이션 업그레이드 .....</b>	<b>7</b>
2.1. MAVEN 종속 항목 정보	7
2.2. FUSE 프로젝트의 MAVEN 종속성 업데이트	7
<b>3장. JBOSS EAP 독립 실행형에서 FUSE 애플리케이션 업그레이드 .....</b>	<b>9</b>
3.1. MAVEN 종속 항목 정보	9
3.2. FUSE 프로젝트의 MAVEN 종속성 업데이트	9
3.3. JBOSS EAP 설치에서 기존 FUSE 업그레이드	10
<b>4장. KARAF 독립 실행형에서 FUSE 애플리케이션 업그레이드 .....</b>	<b>11</b>
4.1. MAVEN 종속 항목 정보	11
4.2. FUSE 프로젝트의 MAVEN 종속성 업데이트	11
<b>5장. KARAF에서 FUSE STANDALONE 업그레이드 .....</b>	<b>13</b>
5.1. FUSE ON KARAF를 업그레이드할 때의 영향	13
5.2. KARAF에서 FUSE STANDALONE 업그레이드	13
5.3. KARAF에서 FUSE의 업그레이드 롤백	15



## 머리말

이 가이드에서는 Red Hat Fuse 및 Fuse 애플리케이션 업데이트에 대한 정보를 제공합니다.



### 참고

이 가이드의 지침에 따라 Fuse 6에서 최신 Fuse 7 릴리스로 마이그레이션하려면 [Red Hat Fuse 7.0 마이그레이션 가이드](#)의 지침을 따라야 합니다.

1장. *Fuse Online* 업그레이드

2장. *Spring Boot* 독립형에서 *Fuse* 애플리케이션 업그레이드

3장. *JBoss EAP* 독립 실행형에서 *Fuse* 애플리케이션 업그레이드

4장. *Karaf* 독립 실행형에서 *Fuse* 애플리케이션 업그레이드

5장. *Karaf*에서 *Fuse Standalone* 업그레이드

# 1장. FUSE ONLINE 업그레이드

Fuse Online 업그레이드 프로세스는 Fuse Online이 Red Hat OpenShift Online 또는 OCP(OpenShift Container Platform)에 설치되어 있는지에 따라 다릅니다.


- **OpenShift Online** - Fuse 7.6이 릴리스되면 OpenShift Online의 Fuse Online 인프라가 자동으로 업그레이드됩니다. [OpenShift Online에서 실행 중인 Upgrading Fuse Online 통합에 설명된 대로 실행 중인](#) 모든 통합을 다시 게시해야 합니다.
- **OCP** - OpenShift Container Platform 온-사이트에서 실행 중인 Fuse Online 환경을 업그레이드하려면 최신 Fuse Online 릴리스를 다운로드하고 업데이트 스크립트를 실행한 다음 [OCP의 Fuse Online](#) 업그레이드에 설명된 대로 실행 중인 통합을 다시 게시해야 합니다.

## 1.1. OPENSIFT ONLINE에서 실행 중인 FUSE ONLINE 통합 업그레이드

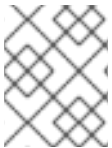
Fuse 7.6이 릴리스되면 OpenShift Online의 Fuse Online 인프라가 자동으로 업그레이드됩니다. 인프라 업그레이드 중에 OpenShift Online에서 실행되는 기존 통합은 업그레이드 중 및 이후에 계속 실행됩니다. 그러나 기존 통합은 이전 버전의 Fuse 라이브러리 및 종속 항목을 계속 실행합니다.

Fuse Online 인프라가 새 릴리스로 업그레이드되었음을 알리는 이메일 메시지가 표시되면 기존 통합을 다시 게시하여 업그레이드하십시오(단일 다시 시작). 가능한 한 빨리 이 작업을 수행합니다.

통합을 다시 게시하려면 왼쪽 탐색 패널에서 Fuse Online 환경에서 통합을 **클릭합니다**. 각 통합에 대해 다음을 수행합니다.

1. 통합 항목 오른쪽에 있는  을 클릭하고 **편집**을 선택합니다.
2. Fuse Online에서 편집하기 위한 통합이 표시되면 오른쪽 상단에 **게시** 를 클릭합니다.

게시하면 최신 Fuse Online 종속성을 사용하는 다시 빌드가 수행됩니다.



### 참고

Fuse Online 사용자 인터페이스에서는 통합 요소에 최신 종속성이 있는 경우 업데이트해야 하는 경고가 표시됩니다.

## 1.2. OCP에서 FUSE ONLINE 업그레이드

OCP에서 Fuse Online을 업그레이드하려면 최신 Fuse Online 릴리스를 다운로드하고, 클러스터 관리자에서 Fuse Online을 업그레이드할 수 있는 권한을 확보하고, 업데이트 스크립트를 실행합니다.

경우에 따라 패치 및 보안 수정 사항을 통합하는 새로운 애플리케이션 이미지가 Fuse Online용으로 릴리스됩니다. Red Hat의 에라타 업데이트 채널을 통해 이러한 업데이트에 대한 알림을 받습니다. 그런 다음 Fuse Online 이미지를 업그레이드할 수 있습니다.

다음 업그레이드의 업그레이드 절차는 동일합니다.

- Fuse Online 7.5에서 Fuse Online 7.6까지
- Fuse Online 7.6 버전에서 최신 Fuse Online 7.6 버전으로 사용

사전 요구 사항



- 사이트에서 OCP Online 버전 7.5를 설치하고 실행 중입니다. 또는 OCP 사이트에서 7.6 버전의 Fuse Online을 설치하고 실행 중이며 새로운 애플리케이션 이미지로 업그레이드하려고 합니다. 이전 버전의 경우:
  - OCP에서 7.4 of Fuse Online 버전을 실행 중인 경우 **7.5로 업그레이드해야** 하며 7.6로 업그레이드할 수 있습니다.
  - OCP에서 Fuse Online 버전 7.3을 실행 중인 경우 **7.4로 업그레이드해야** 하며 7.5로 업그레이드할 수 있습니다.
  - OCP에서 7.2 버전의 Fuse Online을 실행 중인 경우 **7.3 버전으로 업그레이드해야** 합니다.
  - OCP에서 7.1 버전의 Fuse Online을 실행 중인 경우 **7.2로 업그레이드해야** 합니다.
- **oc** 클라이언트 도구를 설치하고 Fuse Online이 설치된 OCP 클러스터에 연결되어 있습니다.
- 이 절차의 처음 5 단계에 필요한 클러스터 관리 권한이 있습니다.

## 절차

1. 클러스터 관리자는 Fuse Online 패키지를 다운로드하고 사용자가 특정 프로젝트에서 Fuse Online을 업그레이드할 수 있는 권한을 부여합니다.
  - a. 다음 위치에서 Fuse Online 설치 스크립트가 포함된 패키지를 다운로드합니다.  
<https://github.com/syndesisio/fuse-online-install/releases/tag/1.9>

파일 시스템의 편리한 위치에 다운로드한 아카이브의 압축을 풉니다. **fuse-online-install-1.9** 디렉터리에는 Fuse Online 업그레이드를 위한 스크립트 및 지원 파일이 포함되어 있습니다.
  - b. 추출된 아카이브가 포함된 디렉터리로 변경합니다. 예를 들면 다음과 같습니다.  
**cd fuse-online-install-1.9**
  - c. 클러스터 관리 계정을 사용하여 OpenShift에 로그인합니다. 예를 들면 다음과 같습니다.  
**oc login -u admin -p admin**
  - d. Fuse Online을 업그레이드해야 하는 OpenShift 프로젝트로 변경합니다. 예를 들면 다음과 같습니다.  
**oc project fuse-online-project**
  - e. 이 프로젝트에서만 Fuse Online 업그레이드 권한을 부여합니다. 예를 들어 다음 명령은 Fuse Online을 **developer** 사용자로 업그레이드할 수 있는 권한을 부여합니다. 클러스터 관리자가 이 명령을 실행한 후 **developer** 사용자는 이 프로젝트에서만 Fuse Online을 업그레이드할 수 있습니다(예: **fuse-online-project** ).  
**bash install\_ocp.sh --grant developer**
2. Fuse Online 업그레이드 권한이 부여된 사용자는 업그레이드를 수행합니다.
  - a. OpenShift에 로그인합니다. 예를 들면 다음과 같습니다.  
**oc login -u developer**
  - b. Fuse Online을 업그레이드하려는 프로젝트로 전환합니다. 예를 들면 다음과 같습니다.  
**oc project fuse-online-project**
  - c. 업그레이드할 버전을 확인하려면 다음과 같이 **--version** 옵션을 사용하여 업데이트 스크립트를 실행합니다.  
**bash update\_ocp.sh --version**

d. 다음과 같이 업데이트 스크립트를 호출합니다.

**bash update\_ocp.sh**

스크립트에 대해 자세히 알아보려면 **bash update\_ocp.sh --help** 를 호출합니다.

인프라 업그레이드 중 및 이후에 기존 통합은 이전 버전의 Fuse Online 라이브러리 및 종속 항목을 계속 실행합니다.

3. 다음과 같이 실행 중인 Fuse Online 통합 업그레이드:

a. Fuse Online에서 업그레이드할 통합을 선택합니다.

b. 편집을 선택합니다.

c. 게시 를 선택하여 통합을 다시 게시합니다.

통합을 다시 게시하면 최신 Fuse Online 종속성을 사용하는 다시 빌드가 수행됩니다.

## 2장. SPRING BOOT 독립형에서 FUSE 애플리케이션 업그레이드

Spring Boot에서 Fuse 애플리케이션을 업그레이드하려면 Fuse 프로젝트의 Maven 종속성을 업데이트하여 올바른 Fuse 버전을 사용해야 합니다.

일반적으로 Maven을 사용하여 Fuse 애플리케이션을 빌드합니다. Maven은 Apache의 무료 오픈 소스 빌드 툴입니다. Maven 구성은 Fuse 애플리케이션 프로젝트의 **pom.xml** 파일에 정의되어 있습니다. Fuse 프로젝트를 빌드하는 동안 기본 동작은 Maven이 외부 리포지토리를 검색하고 필요한 아티팩트를 다운로드하는 것입니다. Maven 빌드 프로세스에서 올바른 Fuse 지원 아티팩트 세트를 선택할 수 있도록 BOM(Fuse of material)에 대한 종속성을 **pom.xml** 파일에 추가합니다.

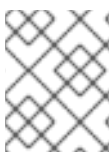
다음 섹션에서는 Maven 종속 항목에 대한 정보와 Fuse 프로젝트에서 업데이트하는 방법을 설명합니다.

- 2.1절. "Maven 종속 항목 정보"
- 2.2절. "Fuse 프로젝트의 Maven 종속성 업데이트"

### 2.1. MAVEN 종속 항목 정보

BOM(Maven bill of materials) 파일의 목적은 잘 작동하는 Maven 종속성 버전 집합을 제공하여 모든 Maven 아티팩트에 대해 버전을 개별적으로 정의할 필요가 없도록 하는 것입니다.

Fuse가 실행되는 각 컨테이너에 전용 BOM 파일이 있습니다.

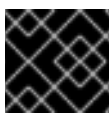


#### 참고

이러한 BOM 파일을 여기에서 찾을 수 있습니다. <https://github.com/jboss-fuse/redhat-fuse>. 또는 BOM 파일 업데이트에 대한 정보는 [최신 릴리스 노트](#) 로 이동합니다.

Fuse BOM은 다음과 같은 이점을 제공합니다.

- **pom.xml** 파일에 종속성을 추가할 때 버전을 지정할 필요가 없도록 Maven 종속 항목에 대한 버전을 정의합니다.
- 특정 버전의 Fuse에서 완전히 테스트 및 지원되는 선별된 종속성 세트를 정의합니다.
- Fuse 업그레이드 간소화.



#### 중요

Fuse BOM에서 정의한 종속성 세트만 Red Hat에서 지원됩니다.

### 2.2. FUSE 프로젝트의 MAVEN 종속성 업데이트

Spring Boot 용 Fuse 애플리케이션을 업그레이드하려면 프로젝트의 Maven 종속성을 업데이트합니다.

#### 절차

1. 프로젝트의 **pom.xml** 파일을 엽니다.
2. 다음 예와 같이 프로젝트의 **pom.xml** 파일(또는 상위 **pom.xml** 파일)에 **dependencyManagement** 요소를 추가합니다.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

```

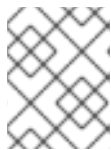
<project ...>
...
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

  <!-- configure the versions you want to use here -->
  <fuse.version>7.6.0.fuse-760027-redhat-00001 </fuse.version>

</properties>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.jboss.redhat-fuse</groupId>
      <artifactId>fuse-springboot-bom</artifactId>
      <version>${fuse.version}</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
...
</project>

```



### 참고

Spring Boot 버전도 업데이트해야 합니다. 일반적으로 **pom.xml** 파일의 Fuse 버전에서 찾을 수 있습니다.

```

<properties>
  <!-- configure the versions you want to use here -->
  <fuse.version>7.6.0.fuse-760027-redhat-00001 </fuse.version>
  <spring-boot.version>1.5.22.RELEASE</spring-boot.version>
</properties>

```

### 3. **pom.xml** 파일을 저장합니다.

**pom.xml** 파일에서 BOM을 종속성으로 지정하면 아티팩트 버전을 지정하지 않고 Maven 종속성을 **pom.xml** 파일에 추가할 수 있습니다. 예를 들어 **camel-velocity** 구성 요소에 대한 종속성을 추가하려면 **pom.xml** 파일의 종속 항목에 다음 XML 조각을 추가합니다.

```

<dependency>
  <groupId>org.apache.camel</groupId>
  <artifactId>camel-velocity</artifactId>
  <scope>provided</scope>
</dependency>

```

이 종속성 정의에서 **version** 요소를 생략하는 방법을 참조하십시오.

## 3장. JBOSS EAP 독립 실행형에서 FUSE 애플리케이션 업그레이드

JBoss EAP에서 Fuse 애플리케이션을 업그레이드하려면 올바른 Fuse 버전을 사용하도록 Fuse 프로젝트의 Maven 종속성을 업데이트해야 합니다.

일반적으로 Maven을 사용하여 Fuse 애플리케이션을 빌드합니다. Maven은 Apache의 무료 오픈 소스 빌드 툴입니다. Maven 구성은 Fuse 애플리케이션 프로젝트의 **pom.xml** 파일에 정의되어 있습니다. Fuse 프로젝트를 빌드하는 동안 기본 동작은 Maven이 외부 리포지토리를 검색하고 필요한 아티팩트를 다운로드 하는 것입니다. Maven 빌드 프로세스에서 올바른 Fuse 지원 아티팩트 세트를 선택할 수 있도록 BOM(Fuse of material)에 대한 종속성을 **pom.xml** 파일에 추가합니다.

다음 섹션에서는 Maven 종속 항목에 대한 정보와 Fuse 프로젝트에서 업데이트하는 방법을 설명합니다.

- 3.1절. "Maven 종속 항목 정보"
- 3.2절. "Fuse 프로젝트의 Maven 종속성 업데이트"

### 3.1. MAVEN 종속 항목 정보

BOM(Maven bill of materials) 파일의 목적은 잘 작동하는 Maven 종속성 버전 집합을 제공하여 모든 Maven 아티팩트에 대해 버전을 개별적으로 정의할 필요가 없도록 하는 것입니다.

Fuse가 실행되는 각 컨테이너에 전용 BOM 파일이 있습니다.

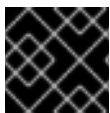


#### 참고

이러한 BOM 파일을 여기에서 찾을 수 있습니다. <https://github.com/jboss-fuse/redhat-fuse>. 또는 BOM 파일 업데이트에 대한 정보는 [최신 릴리스 노트](#) 로 이동합니다.

Fuse BOM은 다음과 같은 이점을 제공합니다.

- **pom.xml** 파일에 종속성을 추가할 때 버전을 지정할 필요가 없도록 Maven 종속 항목에 대한 버전을 정의합니다.
- 특정 버전의 Fuse에서 완전히 테스트 및 지원되는 선별된 종속성 세트를 정의합니다.
- Fuse 업그레이드 간소화.



#### 중요

Fuse BOM에서 정의한 종속성 세트만 Red Hat에서 지원합니다.

### 3.2. FUSE 프로젝트의 MAVEN 종속성 업데이트

JBoss EAP용 Fuse 애플리케이션을 업그레이드하려면 프로젝트의 Maven 종속성을 업데이트합니다.

#### 절차

1. 프로젝트의 **pom.xml** 파일을 엽니다.
2. 다음 예와 같이 프로젝트의 **pom.xml** 파일(또는 상위 **pom.xml** 파일)에 **dependencyManagement** 요소를 추가합니다.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

```

<project ...>
...
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

  <!-- configure the versions you want to use here -->
  <fuse.version>7.6.0.fuse-760027-redhat-00001 </fuse.version>

</properties>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.jboss.redhat-fuse</groupId>
      <artifactId>fuse-eap-bom</artifactId>
      <version>${fuse.version}</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
...
</project>

```

3. **pom.xml** 파일을 저장합니다.

**pom.xml** 파일에서 BOM을 종속성으로 지정하면 아티팩트 버전을 지정하지 않고 Maven 종속성을 **pom.xml** 파일에 추가할 수 있습니다. 예를 들어 **camel-velocity** 구성 요소에 대한 종속성을 추가하려면 **pom.xml** 파일의 종속 항목에 다음 XML 조각을 추가합니다.

```

<dependency>
  <groupId>org.apache.camel</groupId>
  <artifactId>camel-velocity</artifactId>
  <scope>provided</scope>
</dependency>

```

이 종속성 정의에서 **version** 요소를 생략하는 방법을 참조하십시오.

### 3.3. JBOSS EAP 설치에서 기존 FUSE 업그레이드

다음 절차에서는 JBoss EAP 설치에서 기존 Fuse를 업그레이드하는 방법을 설명합니다.

#### 절차

1. 하나의 JBoss EAP 마이너 릴리스에서 다른 릴리스로 업그레이드하려면 [JBoss EAP 패치 및 업그레이드 가이드 가이드](#)의 지침을 따라야 합니다.
2. Fuse를 업데이트하려면 JBoss EAP에 설치 가이드에 설명된 대로 [Fuse on JBoss EAP 설치](#) 프로그램을 실행해야 합니다.



#### 참고

Fuse 애플리케이션을 다시 컴파일하거나 축소할 필요가 없습니다.

## 4장. KARAF 독립 실행형에서 FUSE 애플리케이션 업그레이드

Karaf에서 Fuse 애플리케이션을 업그레이드하려면 Fuse 프로젝트의 Maven 종속성을 업데이트하여 올바른 Fuse 버전을 사용해야 합니다.

일반적으로 Maven을 사용하여 Fuse 애플리케이션을 빌드합니다. Maven은 Apache의 무료 오픈 소스 빌드 툴입니다. Maven 구성은 Fuse 애플리케이션 프로젝트의 **pom.xml** 파일에 정의되어 있습니다. Fuse 프로젝트를 빌드하는 동안 기본 동작은 Maven이 외부 리포지토리를 검색하고 필요한 아티팩트를 다운로드하는 것입니다. Maven 빌드 프로세스에서 올바른 Fuse 지원 아티팩트 세트를 선택할 수 있도록 BOM(Fuse of material)에 대한 종속성을 **pom.xml** 파일에 추가합니다.

다음 섹션에서는 Maven 종속 항목에 대한 정보와 Fuse 프로젝트에서 업데이트하는 방법을 설명합니다.

- 4.1절. "Maven 종속 항목 정보"
- 4.2절. "Fuse 프로젝트의 Maven 종속성 업데이트"

### 4.1. MAVEN 종속 항목 정보

BOM(Maven bill of materials) 파일의 목적은 잘 작동하는 Maven 종속성 버전 집합을 제공하여 모든 Maven 아티팩트에 대해 버전을 개별적으로 정의할 필요가 없도록 하는 것입니다.

Fuse가 실행되는 각 컨테이너에 전용 BOM 파일이 있습니다.

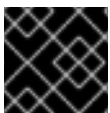


#### 참고

이러한 BOM 파일을 여기에서 찾을 수 있습니다. <https://github.com/jboss-fuse/redhat-fuse>. 또는 BOM 파일 업데이트에 대한 정보는 [최신 릴리스 노트](#) 로 이동합니다.

Fuse BOM은 다음과 같은 이점을 제공합니다.

- **pom.xml** 파일에 종속성을 추가할 때 버전을 지정할 필요가 없도록 Maven 종속 항목에 대한 버전을 정의합니다.
- 특정 버전의 Fuse에서 완전히 테스트 및 지원되는 선별된 종속성 세트를 정의합니다.
- Fuse 업그레이드 간소화.



#### 중요

Fuse BOM에서 정의한 종속성 세트만 Red Hat에서 지원합니다.

### 4.2. FUSE 프로젝트의 MAVEN 종속성 업데이트

Karaf용 Fuse 애플리케이션을 업그레이드하려면 프로젝트의 Maven 종속성을 업데이트합니다.

#### 절차

1. 프로젝트의 **pom.xml** 파일을 엽니다.
2. 다음 예와 같이 프로젝트의 **pom.xml** 파일(또는 상위 **pom.xml** 파일)에 **dependencyManagement** 요소를 추가합니다.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

```

<project ...>
...
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

  <!-- configure the versions you want to use here -->
  <fuse.version>7.6.0.fuse-760027-redhat-00001 </fuse.version>

</properties>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.jboss.redhat-fuse</groupId>
      <artifactId>fuse-karaf-bom</artifactId>
      <version>${fuse.version}</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
...
</project>

```

### 3. pom.xml 파일을 저장합니다.

**pom.xml** 파일에서 BOM을 종속성으로 지정하면 아티팩트 버전을 지정하지 않고 Maven 종속성을 **pom.xml** 파일에 추가할 수 있습니다. 예를 들어 **camel-velocity** 구성 요소에 대한 종속성을 추가하려면 **pom.xml** 파일의 종속 항목에 다음 XML 조각을 추가합니다.

```

<dependency>
  <groupId>org.apache.camel</groupId>
  <artifactId>camel-velocity</artifactId>
  <scope>provided</scope>
</dependency>

```

이 종속성 정의에서 **version** 요소를 생략하는 방법을 참조하십시오.



## 5장. KARAF에서 FUSE STANDALONE 업그레이드

Fuse on Apache Karaf 업그레이드 메커니즘을 사용하면 Karaf에서 업데이트된 버전의 Fuse를 다시 설치할 필요 없이 Apache Karaf 컨테이너에 수정 사항을 적용할 수 있습니다. 또한 업그레이드로 인해 배포된 애플리케이션에 문제가 발생하는 경우 업그레이드를 롤백할 수 있습니다.

업그레이드 설치 프로그램 파일은 Apache Karaf에 Fuse를 설치하는 데 사용하는 것과 동일한 파일입니다.



### 참고

업그레이드 설치 프로그램 파일을 얻으려면 Red Hat 고객 포털의 [다운로드 페이지](#)로 이동하여 Apache Karaf에서 Fuse용 최신 버전의 설치 아카이브(예: **fuse-karaf-7.6.0.fuse-760025-redhat-00001.zip**)를 다운로드합니다.

- 5.1절. "Fuse on Karaf를 업그레이드할 때의 영향"
- 5.2절. "Karaf에서 Fuse Standalone 업그레이드"
- 5.3절. "Karaf에서 Fuse의 업그레이드 롤백"

### 5.1. FUSE ON KARAF를 업그레이드할 때의 영향

업그레이드 메커니즘은 **변들 JAR 및 정적 파일(예: etc/ 디렉토리 아래의 구성 파일 포함)**을 포함한 모든 설치 파일을 업데이트할 수 있습니다. Apache Karaf 업그레이드 프로세스의 Fuse:

- 변들 JAR, 구성 파일 및 정적 파일을 포함한 모든 파일을 업데이트합니다.
- 현재 컨테이너 인스턴스(및 **데이터/** 디렉토리 아래의 런타임 스토리지)와 기본 설치를 모두 패치합니다. 따라서 컨테이너 인스턴스를 삭제한 후 패치가 유지됩니다.
- 기능 리포지토리 파일 및 기능 자체를 포함하여 Karaf 기능과 관련된 모든 파일을 업데이트합니다. 따라서 롤업 패치 후에 설치된 모든 기능은 올바른 패치된 종속 항목을 참조합니다.
- 필요한 경우 구성 파일(예: **etc/**아래 파일)을 업데이트하고 패치를 통해 변경한 구성 변경 사항을 자동으로 병합합니다. 병합 충돌이 발생하는 경우 패치 로그에서 처리 방법에 대한 자세한 내용을 참조하십시오.
- 대부분의 병합 충돌은 자동으로 해결됩니다. 예를 들어 패치 메커니즘은 속성 파일의 속성 수준에서 충돌을 감지합니다. 사용자인지 또는 속성을 변경한 패치인지 여부를 감지합니다. 한 쪽만 속성을 변경한 경우 변경 사항은 보존됩니다.
- 패치를 롤백할 수 있도록 설치에 수행된 **모든 변경 사항(정의 파일 포함)**을 추적합니다.



### 참고

롤업 패치 메커니즘은 내부 git 리포지토리( **patches/.management/history**)를 사용하여 변경 사항을 추적합니다.

### 5.2. KARAF에서 FUSE STANDALONE 업그레이드

다음 지침에서는 Apache Karaf에서 Fuse를 업그레이드하는 방법을 설명합니다. 업그레이드 절차를 시작하기 전에 모든 사전 요구 사항이 완료되었는지 확인합니다.

#### 사전 요구 사항

- 업그레이드하기 전에 Apache Karaf 설치에서 Fuse를 전체 백업해야 합니다.
- 아직 실행되지 않은 경우 컨테이너를 시작합니다.

## 작은 정보

컨테이너가 백그라운드에서 실행 중인 경우(또는 원격으로) SSH 콘솔 클라이언트인 **bin/client** 를 사용하여 컨테이너에 연결합니다.

- **patch:add** 명령을 호출하여 업그레이드 설치 프로그램 파일을 컨테이너의 환경에 추가합니다. 예를 들어 **fuse-karaf-7.6.0.fuse-760025-redhat-00001.zip** 업그레이드 설치 프로그램 파일을 추가하려면 다음을 수행합니다.

```
patch:add file:///path/to/fuse-karaf-7.6.0.fuse-760025-redhat-00001.zip
```

## 절차

1. **patch:update** 명령을 실행합니다. 컨테이너를 다시 시작할 필요가 없습니다.

```
karaf@root(>) patch:update
Current patch mechanism version: 7.1.0.fuse-710023-redhat-00001
New patch mechanism version detected: 7.2.0.fuse-720035-redhat-00001
Uninstalling patch features in version 7.1.0.fuse-710023-redhat-00001
Installing patch features in version 7.2.0.fuse-720035-redhat-00001
```

2. **patch:list** 명령을 호출하여 업그레이드 설치 프로그램 목록을 표시합니다. 이 목록에서 **[name]** 제목 아래의 항목은 업그레이드 ID입니다. 예를 들면 다음과 같습니다.

```
karaf@root(>) patch:list
[name] [installed] [rollup] [description]
fuse-karaf-7.2.0.fuse-720035-redhat-00001 false true fuse-karaf-7.2.0.fuse-720035-
redhat-00001
```

3. **patch:simulate** 명령을 호출하고 적용할 업그레이드 ID를 다음과 같이 지정하여 업그레이드를 시뮬레이션합니다.

```
karaf@root(>) patch:simulate fuse-karaf-7.2.0.fuse-720035-redhat-00001
INFO : org.jboss.fuse.modules.patch.patch-management (226): Installing rollup patch "fuse-
karaf-7.2.0.fuse-720035-redhat-00001"
===== Repositories to remove (9):
- mvn:io.hawt/hawtio-karaf/2.0.0.fuse-710018-redhat-00002/xml/features
...
===== Repositories to add (9):
- mvn:io.hawt/hawtio-karaf/2.0.0.fuse-720044-redhat-00001/xml/features
...
===== Repositories to keep (10):
- mvn:org.apache.activemq/artemis-features/2.4.0.amq-711002-redhat-1/xml/features
...
===== Features to update (100):
[name] [version] [new version]
aries-blueprint 4.2.0.fuse-710024-redhat-00002 4.2.0.fuse-720061-redhat-00001
...
===== Bundles to update as part of features or core bundles (100):
[symbolic name] [version] [new location]
```

```

io.hawt.hawtio-log                2.0.0.fuse-710018-redhat-00002
mvn:io.hawt.hawtio-log/2.0.0.fuse-720044-redhat-00001
...
===== Bundles to reinstall as part of features or core bundles (123):
[symbolic name]                [version]                [location]
com.fasterxml.jackson.core.jackson-annotations    2.8.11
mvn:com.fasterxml.jackson.core/jackson-annotations/2.8.11
...
Simulation only - no files and runtime data will be modified.
karaf@root(>)

```

이렇게 하면 업그레이드가 수행되지만 컨테이너를 실제로 변경하지 않는 경우 컨테이너에 대한 변경 로그가 생성됩니다. 시뮬레이션 로그를 검토하여 컨테이너에 적용할 변경 사항을 파악합니다.

4. **patch:install** 명령을 호출하여 컨테이너를 업그레이드하고 적용할 업그레이드 ID를 지정합니다. 예를 들면 다음과 같습니다.

```
karaf@root(>) patch:install fuse-karaf-7.6.0.fuse-760025-redhat-00001
```

5. 업그레이드 아티팩트 중 하나를 검색하여 업그레이드를 확인합니다. 예를 들어 Fuse 7.1.0을 Fuse 7.2.0으로 업그레이드한 경우 다음과 같이 빌드 번호 760032로 번들을 검색할 수 있습니다.

```

karaf@root(>) bundle:list -l | grep 760032
 22 | Active | 80 | 7.6.0.fuse-760025-redhat-00001 | mvn:org.jboss.fuse.modules/fuse-
pax-transx-tm-narayana/7.6.0.fuse-760025-redhat-00001
188 | Active | 80 | 7.6.0.fuse-760025-redhat-00001 |
mvn:org.jboss.fuse.modules.patch/patch-commands/7.6.0.fuse-760025-redhat-00001

```



## 참고

업그레이드 후 컨테이너를 다시 시작할 때 welcome 배너에 새 버전과 빌드 번호도 표시됩니다.

## 5.3. KARAF에서 FUSE의 업그레이드 롤백

경우에 따라 업그레이드가 작동하지 않거나 컨테이너에 새로운 문제가 발생할 수 있습니다. 이 경우 **patch:rollback** 명령을 사용하여 업그레이드를 쉽게 롤백하고 시스템을 이전 상태로 복원할 수 있습니다. 이 지침에서는 이 절차를 안내합니다.

### 사전 요구 사항

- 최근 Karaf에서 Fuse를 업그레이드했습니다.
- 업그레이드를 롤백하려고 합니다.

### 절차

1. **patch:list** 명령을 호출하여 가장 최근에 설치된 패치의 업그레이드 ID **UPGRADE\_ID** 를 가져옵니다.
2. 다음과 같이 **patch:rollback** 명령을 호출합니다.

```
patch:rollback UPGRADE_ID
```



## 참고

경우에 따라 업그레이드를 롤백하기 위해 컨테이너를 다시 시작해야 합니다. 이 경우 컨테이너가 자동으로 다시 시작됩니다. OSGi 런타임의 매우 동적인 특성으로 인해 재시작 중에 호환되지 않는 클래스와 관련된 일부 오류가 표시될 수 있습니다. 이러한 오류는 방금 시작하거나 중지하여 무시해도 되는 OSGi 서비스와 관련이 있습니다.